Ego-motion Analysis Using Average Image Data Intensity

Kojiro Kato
The University of Electro-Communications
1-5-1 Chofugaoka
Chofu, Tokyo, Japan
kojiro@vogue.is.uec.ac.jp

Kris M. Kitani
The University of Electro-Communications
1-5-1 Chofugaoka
Chofu, Tokyo, Japan
kitani@is.uec.ac.jp

Takuya Nojima
The University of Electro-Communications
1-5-1 Chofugaoka
Chofu, Tokyo, Japan
tnojima@computer.org

ABSTRACT

In this paper, we present a new method to perform ego-motion analysis using intensity averaging of image data. The method can estimate general motions from two sequential images on pixel plane by calculating cross correlations. With distance information between camera and objects, this method also enables estimates of camera motion. This method is sufficiently robust even for out of focus image and the calculational overhead is quite low because it uses a simple averaging method. In the future, this method could be used to measure fast motions such as human head tracking, or robot movement. We present a detailed description of the proposed method, and experimental results demonstrating its basic capability. With these results, we verify that our proposed system can detect camera motion even with blurred images. Furthermore, we confirm that it can operate at up to 714 FPS in calculating one dimensional translation motion.

Categories and Subject Descriptors

I.4.8 [**Scene Analysis**]: Motion, Intensity, color, photometry, and thresholding

General Terms

Algorithms, Measurement, Experimentation, Human Factors, and Verification.

Keywords

Ego-motion estimation, Image processing, Correlation, Averaging image.

1. INTRODUCTION

Given the progress of motion capture technology, especially in the field of image recognition, human motion is now possible as a computer interface, for instance, Wii Remote (Nintendo, co, Ltd) detects human motion by using controller equipped the accelerometer, the gyro sensor, and the camera. A camera on the Wii Remote captures the Infrared LED markers installed on a sensor bar, and then calculates relative motion between the sensor bar and the hand-held camera (i.e. Wii Remote itself). Kinect (Microsoft) , consisting of an infrared projector for depth sensing and cameras for motion analysis, recognizes a user's movement in its view. In these examples, there are two kinds of techniques employed in motion analysis using image data. One is fixing cameras in the environment and tracks the user s' motion. This configuration is often used in conventional motion capture systems like OptiTrack(NaturalPoint, Inc),Vicon(Vicon motion systems), etc. Those systems can measure multiple markers on the body, or general motion of human body through motion analysis based on the camera images. However, it is hard for such systems to measure users' motion that walk around wide area because cameras should be fixed to the environment. If their body comes off from the viewing area, those systems cannot estimate their motion.

The other technique is mounting cameras on users' body and measures the part of body motion through estimating ego-motion analysis of camera. This configuration is not common as the cameras-in-the-environment system mentioned above. Typical examples are Hiball[1] and Wii Remote. Those systems are equipped with photo sensitive device or camera. Markers are placed on the environment and those systems detect relative motion between the markers and the system itself. They are mounted on user's body, or user hold the device then the system can estimate motion information of the specific part of the body. This means, that those systems could not perceive limited motion information. Adding to that, they have same limitation on sensing area as mentioned in the cameras-in-the-environment system. They cannot detect motion out of the marker available area. To solve this problem, ego-motion estimation through optical flow of images is often used. Two representative techniques in calculating optical flow are block matching and gradient-based methods. Block matching can calculate optical flows if camera movements are large, but it is unable to be used for human motion capture, as the computation speed is too slow and the method have a problem of the parallax. By comparison, gradient-based method is faster calculating optical flow. However, it also is unable to be used for human motion, because of the slow calculational overhead.

Cameras-on-the-body technique without markers is appropriate detecting motion of objects that moves in a wide area such as around human. In this research, we focus on a method that involves body-mounted cameras. The calculational overhead of conventional method is high for real-time measurement of motion. Furthermore, estimating motion is hard for such method when the image is out of focus.

We show that by calculating cross-correlations of intensity averages between two sequential images the method is able to

AH '11, Mar 13-13 2011, Tokyo, Japan

track human motion very easily and quickly. Then we will discuss on suitable applications.

2. RELATED WORK

In this section, the ego-motion estimation technique relevant to image data processing is briefly described.

Marker based ego-motion estimation is often used for capturing human motion. Greg *et. al.* have proposed a high-speed head tracking system named Hiball[1]. The user of this system mounts capture device on their head. Then that device captures infrared active markers installed on ceiling to calculate user's head motion. Akito *et. al.* have proposed a pointing system by using multiple transparent markers and a camera[3]. They use ARToolkit[2] to estimate position and orientation of the camera, which they use as a pointing device for remotely placed large wall display. Marker based methods are often used to improve robustness and accuracy of ego-motion estimation, and to reduce calculative cost. However, markers must be prepared before the measurement and sensing area is limited to the area.

The optical computer mouse is a typical example to which this method has been applied. Its motion of the mouse is calculated by comparing the sequential images taken with an image sensor installed in the mouse, it then outputs relative translational motion. The sensor is often used to detect motion. Tanaka[4] has used them to control a mobile robot. However, with this technique, the difficulty lies in that it is only able to track movement in the horizontal plane, incapable of measuring human motions which are intrinsically three-dimensional.

Speed measurement methods using optical flow are being actively researched as a technique in estimating the movement of a camera without tracking markers. For example, Xiaojing[5] estimated a mobile robot's speed considered slippage of a wheel using optical flow calculated from brightness and luminosity, and data obtained from the rotary encoder attached to the wheel. However, optical flow generally entails high calculational overhead.

We can extract space-related information after taking image average. Shibuya *et. al*. proposed a method to estimating a mobile robot's position by using averages and standard deviations of captured images[6]. They compared those data from prepared images and the same from a current image for positional estimation. The calculational overhead of their system is low, although they had to prepare images for the comparison prior to operate the robot. In contrast, the technique that is proposed in this paper is estimate body motions rapidly and without prior data preparation.

In this paper, we propose a method which can calculate speed of camera rapidly to estimate human motion without using feature points.

3. MOTION ANALYSIS METHOD

In our method, we first sum up the image intensity in one direction from a given frame (the vertical direction in case of Figure 1). We can then convert the two dimensional data into a single dimensional form corresponding to an intensity array as described in Figure 1. Here, x axis is the horizontal axis, y axis is the vertical axis, and the intensity of the pixel position is I(x, y). $A_n(x)$ denotes the average intensity of image.

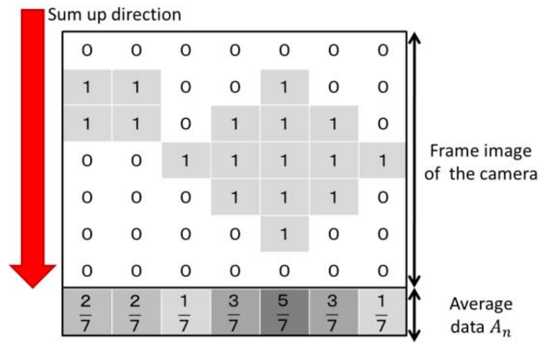


Figure 1. Averaging camera image

Next, correlations of this intensity array are taken between each frame as shown in eq. 1, where R(n) is determined from intensity averages of the current image $A_n(x)$ and of the previous frame $A_{n-1}(x)$.

$$R(j) = \begin{cases} \sum_{i=0}^{N_x-(j+1)} A_n(i) \cdot A_{n-1}(i-j) & (j \geqq 0) \\ R(-j) & (j < 0) \end{cases} \qquad \textbf{eq. 1}$$

N_x : the number of pixels in the x axis direction of the image.

After this procedure, we can estimate the distance between pixels of the intensity distribution for successive frames (D in eq. 2). That distance of pixels contains the information related to the motion of the camera (horizontal motion in case of Figure 1). Finally with the information of distance from the camera to the object, we can estimate the camera motion in physical world.

$$D = [j \text{ such that } R(j) \text{ is maximal}] \quad \text{If and only if } -Nx \leqq j \leqq Nx \qquad \textbf{eq. 2}$$

And, the movement of a camera M_{th} is shown by using size of the photo sensor S_p, size of the image S_i, the distance from the camera to a wall w, and the focal length f like eq. 3. In this formulization, we only consider camera's one-dimensional translational motion, which is parallel to the image sensor. Ego-motion estimation on the rest of the dimensions such as rotation will discussed in discussion section.

$$M_{th} = \frac{S_p}{f \times S_i} \times \frac{D}{W} \qquad \textbf{eq. 3}$$

4. EXPERIMENTS

For study a feasibility of our method, we limit the camera motion to one direction of translation then apply the proposed method. Figure 2 shows the experimental setup.

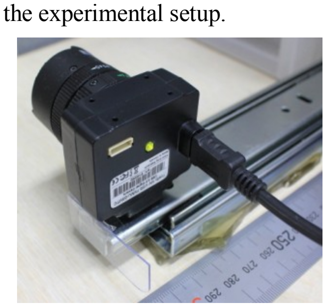


Figure 2.Setup of system.

As shown in the figure, a camera is fixed on the rail. The camera (FireflyMV: FMVU-03MTC-C with operation speed: 63FPS and the maximum resolution: 752pix*480pix and image sensor: 1/3inch) equipped with a lens (T0412FICS-3) of focal length 4mm was used

4.1 Relationship between actual camera motion and pixel displacement

In our method, we assume that pixel displacement will increase in proportion to actual camera motion. To find this, we moved the camera a prescribed distance, and the distance D on the screen by the proposed method. The result is shown in Figure 4. In the figure, horizontal axis means the actual movement of camera [mm] and vertical axis means distance in image plane [pix]. In this experiment, we move the camera along the rail at of 10, 30, 50, and 100[mm] from the original point. Then take pictures five times at each position. From this result, we can deploy the assumption that there is proportional relationship between pixel displacement and actual camera translational motion.

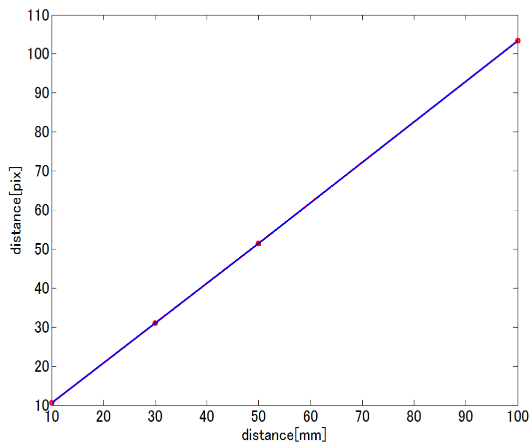


Figure 4. Relationship between actual movement of camera and displacement of image plane (w = 50[mm])

4.2 Influence validation of distance to photographic subject

The relationship between pixel displacement and actual camera motion is strongly affected by the distance between the camera and the objects. The pixel displacement becomes small when the distance become large. To verify this effect, we calculate the pixel displacement while changing the distance between objects and the camera. In this experiment, the actual camera motion is fixed to 50[mm]. Then compare the pixel displacement according to the distance. The result is shown in Figure 3. In the figure, the horizontal axis shows the distance between objects and the camera. The vertical axis indicates pixel displacement calculated through the proposed method. A blue line is theoretical values calculated by (eq. 3) (f = 4[mm], S_p = 4.8[mm], S_i = 640[pix]). The experiment value is almost the same as the theoretical value. It is thought that some of the reasons for the error are measurement error of distance and lens calibration.

From the experimental results described in the chapter 4.1 and 4.2, the proposed method has an ability to estimate ego-motion. Then we presumed the characteristics on the robustness for out of focus and on the low calculational overhead for our proposed method. Experiments to verify are described in the following section.

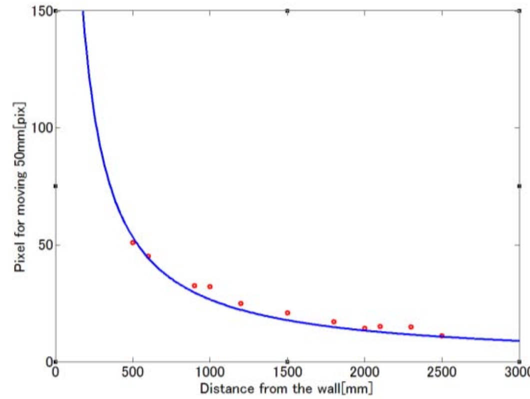
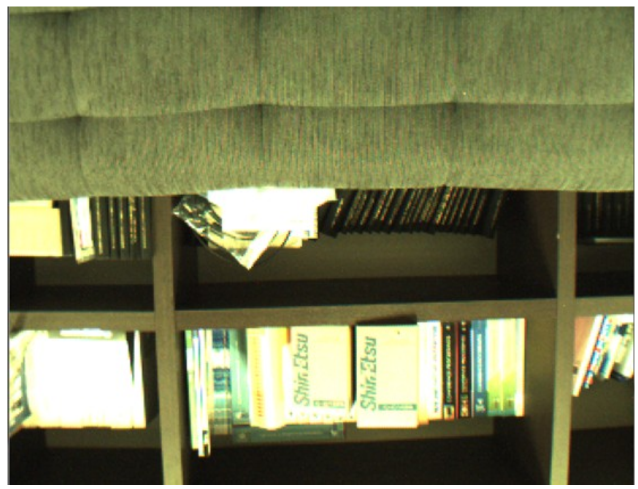


Figure 3. Relationship between movement of image while the camera move 50[mm] and a distance from the wall.



(a) normal condition

(b) out of focus condition

Figure 5. Images of each condition.

4.3 The robustness for out of focus

Our method averages the image. That means, we can use images that are not sharply focused. When mounting cameras on moving objects, the distance between the camera and environment changes dynamically according to the objects' motion. That condition is often hard for conventional method. If our proposed method is robust for the defocusing condition, the movement of a camera can be estimate whether the images has a lot of characteristic or not.

In this section, we conduct an experiment to verify the effect of defocusing. The camera fixed on the rail is placed in front of the bookshelf (Figure 5(a)). The rail is placed parallel to the shelf surface. The approximate distance between the camera and the shelf is 135mm. For the first, we move the camera that is sharply focused at 10/30/50/100 mm. Then we set the camera to defocusing condition (Figure 5(b)) and do the same operation. The result of these experiments is shown in Table.1. From the table, the calculated pixel displacement is approximately the same in spite of the conditional difference on image focus. More detailed experiment is necessary, but the result shows that the proposed method is robust enough for the defocusing condition.

Table 1. Conditions and distance between images [pix].

Condition	movements[mm]			
	10	30	50	100
Normal	5	14.6	22.8	44.6
Defocusing	1	14.8	23	45.4

4.4 Calculational overhead

Finally we evaluate the calculational overhead for this method. In this experiment, we focus on the pure cost of our method. Then we do not consider the cost for image capture, and hardware limitation of frame rate of the camera.

Before the experiment, we record the motion movie through the camera fixed on the rail. Then, we extract that data to the memory in the computer and calculated the distance as early as possible by using the proposed method. We check the maximum frame rate that the algorithm can tolerate. The specification of the computer that we use in this experiment is shown as follows: Model:ThinkPad, CPU:Intel Core2 Duo T9600 2.80GHz, Main Memory: 4GB. The result is shown in Table.2. In this experiment, we change the direction for taking average in 3 cases (x axis, y axis, and diagonal axis). The maximum frame rate of calculating the camera distance is 714FPS when we take average along x axis. Consequently, this technique can calculate rapidly, and apply to a high-speed camera. However, the result shows that the direction for taking average affects the calculational overhead.

Table 2. Computational speed according to the direction.

Direction of calculation	Frames per second
x axis	311.328
y axis	714.173
x axis and y axis	288.996

5. DISCUSSION

By performing these four experiments, we are able to conclude that the proposed method can calculate robustly the movement of the camera from distances between pixels in an image plane for defocusing. Those conclusions confirm that our technique to estimate camera speeds by calculating cross-correlations of intensity averages of sequential images is viable. In addition, computing speeds are very fast supporting the claim that the method would be applicable to estimate human fast actions.

Moreover, it is necessary to experiment further because one must take into consideration the influence of range to the photographic subject and whether picture object is non-planar.

The proposed method has difficulty in calculating translational and rotational motion information independently. By changing averaging axis, the method can measure multiple direction of motion. However, that result is mixture of translational and rotational motion information. Multiple cameras or a camera with other sensors such as gyro sensors are necessary to calculate those information independently.

6. CONCLUSIONS

We have proposed a technique applicable to ego-motion analysis that uses image intensity averages, and performed basic real-time evaluations. Furthermore, by experimental evaluations, it was shown that the proposed technique was able to estimate camera movement at high-speed. However, further evaluation is necessary for figuring out the ability of the proposed method. Then in the future, we will develop a 6-DOF motion measurement system based on this method.

This method should suitable for capturing human motion in moving vehicle such as cars and airplanes. Many research on display systems to improve drivers' and pilots' ability on situation awareness have done these days[7]. In those research that use HMD as display systems often have difficulty installing conventional motion capture system for head tracking because of cost and short of space. Motion captures based on inertial sensors do not require such space but it has difficulty in measuring human and vehicle motion independently. Our proposed method can directly measure relative motion of human against the vehicle. Then in the future, we will conduct research on motion capture system especially for vehicle drivers based on this method.

7. REFERENCES

[1] Welch, G., Bishop, G., Vicci, L., Brumback, S., Keller, K., Colucci, D. 1999. The HiBall tracker: High-Performance Wide-Area Tracking for Virtual and Augmented Environments. In *Proceedings of the ACM symposium on Virtual reality software and technology*.

[2] Kato, H. and Billinghurst, M. 1999. Marker tracking and HMD calibration for a video-based augmented reality conferencing system. In *Proceedings of the 2nd International Workshop on Augmented Reality.*

[3] Hyakutake, A., Ozaki, K., Kitani, K.M., Koike, Hideki. 2010, 3-D interaction with a large wall display using transparent markers. In *Proceedings of the International Conference on Advanced Visual Interface.*

[4] Tanaka, T. 2008. Control of a Moving Robot by Using an Optical Mouse Sensor. In *Research bulletin of Fukuoka Institute of Technology* 40, 2, 217-221.

[5] Song, X., Seneviratne, L.D., Althoefer, K., Song, Z., Vahed, S. M. 2008. A Vision-Based Technique for Vehicle Slip and Velocity Estimation. In *Proceedings of the 17th World Congress The International Federation of Automatic Control* 9215-9220.

[6] Shibuya, N., Umeda, K. 2006. Self-Localization of a Mobile Robot Using Compressed Image Data of Average and Standard Deviation. In *Proceedings of the 18th International Conference on Pattern Recognition*, 4, 614-617.

[7] Funabiki, K., Tsuda, H., Iijima, T., Nojima, T., Tawada, K ., Yoshida, T. 2009. Flight experiment of pilot display for search-and-rescue helicopter. In *Proceedings of SPIE*, 7326.